DOSE FACTORS FOR NON-CAP88 RADIONUCLIDES

Purpose This Air Quality Group procedure describes the steps used to derive dose

factors for non-CAP88 radionuclides.

Scope

This procedure applies to the Rad-NESHAP project and New Source Review Project.

In this procedure

This procedure addresses the following major topics:

Topic	See Page
General Information About This Procedure	2
Who Requires Training to This Procedure?	2
Dose Factors for Non-CAP88 Radionuclides	4
Appendix E, Table 2 Method	5
Federal Guidance Report Method	6
Raw Data Method	8
Checks and Peer Review	21
Records Resulting from This Procedure	22

Hazard Control Plan

The hazard evaluation associated with this work is documented in HCP-ESH-17-Office Work.

Signatures

Prepared by:	Date:
Keith W Jacobson, ESH-17	10/25/2001
Approved by:	Date:
Dave Fuehne, Rad-NESHAP Project Leader	10/26/2001
Approved by:	Date:
Terry Morgan, QA Officer	10/26/01
Work authorized by:	Date:
Doug Stavert, ESH-17 Group Leader	10/28/01

11/05/01

CONTROLLED DOCUMENT

General information about this procedure

Attachments

This procedure has the following attachments:

		No. of
Number	Attachment Title	pages
1	Flow Chart of Process to Calculate Non-CAP88	3
	Radionuclide Doses	
2	Flow Chart of Steps to Calculate Non-CAP88	1
	Radionuclide Doses from Nuclear Decay Data	
3	PERL 5 Programs Used	14
4	Check Sheet for Dose Factors from Nuclear Decay Data	1
5	Quality Assurance Review of Dose Factors Calculated	1
	from Nuclear Decay Data	

History of revision

This table lists the revision history and effective dates of this procedure.

Revision	Date	Description Of Changes
0	11/01/01	New document derived from parts of ESH-17-511,
		R0.

Who requires training to this procedure?

The following personnel require training before implementing this procedure:

• Health physicist assigned to calculate mrem/Ci factors

Training method

The training method for this procedure is **on-the-job training** and is documented in accordance with the procedure for training (ESH-17-024).

Prerequisites

In addition to training to this procedure, the following training is also required prior to performing this procedure:

- Experience or education in the field of health physics
- ESH-17-501, "Dose Assessment Using CAP88"

General information, continued

Definitions specific to this procedure

<u>'pl' scripts:</u> A PERL5 programs used to perform many of the necessary calculations to obtain the necessary dose factors.

References

The following documents are referenced in this procedure:

- ESH-17-024, "Personnel Training"
- ESH-17-501, "Dose Assessment Using CAP88"
- ESH-17-511, "Calculating mrem per Ci Factors"
- DOE Memo for Record: LAAME:6SF-023, Aug 18, 1998
- ESH-17:99-506, Letter to Mr. Steve Fong, Nov 24, 1999
- SEECAL 2.0, "Program to Calculate Age-Dependent Specific Effective Energies", Oak Ridge National Laboratory/Radiation Shielding Information Center CCC-620, May 1995
- EPA 520/1-89-001 "Background Information Document Procedures Approved for Demonstrating Compliance with 40 CFR 61, Subpart I" October 1989
- DOE letter, U.S. DOE Office of the Environment, from Mr. Stephen C. Fong to U.S. EPA Radiation Program Manager, Mr. George P. Brozowski, RE: Addition of Three Radionuclides to CAP88, Aug 18, 1998
- DOE letter, U.S. DOE Office of the Environment, from Mr. Stephen C. Fong to U.S. EPA Radiation Program Manager, Mr. George P. Brozowski, December 9, 1999.
- EPA letter, U.S. EPA Office of Radiation and Indoor Air, from Mr. Frank Marcinowski III, June 26, 2000

Note

Actions specified within this procedure, unless preceded with "should" or "may," are to be considered mandatory guidance (i.e., "shall").

Dose Factors for Non-CAP88 Radionuclides

Background

The Subpart H to 40 CFR 61 states that the emission of any radionuclide (other than radon-222 and radon-220) from DOE facilities is applicable to the dose standard. However, there are many radionuclides encountered at LANL that are not part of the approved dose assessment code, CAP88. This procedure describes how dose factors for non-CAP88 radionuclides are calculated.

EPA approval The EPA has determined that the use of dose factors for radionuclides not originally included in CAP88 require EPA approval. In August of 1998 ESH-17 requested EPA approval of one of the four methods ESH-17 had previously developed (and had been using since 1990) for calculating such doses (DOE letter, 1998). In July of 1999, LANL provided a technical presentation of methods described in this procedure to various representatives of the EPA.

> In December 1999 ESH-17 requested formal approval of all methods used by ESH-17 to calculate doses from non-CAP88 radionuclides (ESH-17, 1999). On June 26, 2000 EPA granted interim approval of one of the methods employed by LANL to calculate dose factors. However, this approval does not cover all non-CAP88 radionuclides. As a solution, the regional EPA office asked that when non-CAP88 radionuclides are reported in the annual Emissions Report, that we report the dose factors used and retain sufficient documentation on the methods employed to derive them.

Determine method to use to calculate doses

There are three different methods described in the procedure used to calculate doses from non-CAP88 radionuclides, each described in a separate chapter of this procedure:

- EPA Table 2 method
- Federal Guidance Report method
- Raw data method

The first method listed above is the easiest and is preferred. The last method is the most difficult.

A fourth method ("Addition of dose factors to CAP88") is described in a letter from DOE to EPA (see references).

Follow the logic of the flow chart (Attachment 1) to determine which method to use. After the doses are determined by one of the above methods, proceed to the chapter Checks and Peer Review.

Appendix E, Table 2 Method

Appendix E, Table 2 Method

Of the three methods presented in this procedure, the most straightforward method is the air concentration-ratio method. This method applies the dose factors given in Appendix E, Table 2 of 40 CFR 61 subpart H. This table has listed many additional radionuclides that are not in CAP88. It should be noted that the mrem per Ci process (given later) has a set of scripts and programs that electronically apply this method.

Steps to calculate dose

To manually calculate a dose, perform the following steps:

Step	Action
1	Prepare to run CAP88 for the particular source in question; the option
	flag #8 (in prepnpt) must be set to "0" to generate the concentration
	tables in the CAP88 output.
2	Enter the amount emitted in the radionuclides section of the prepnpt
	file.
3	In the modifications section of the prepnpt file, be sure to enter the physical decay constant (lamrr) of the radionuclide being modeled, the units should be in d ⁻¹ , e.g.
	\$modi nuc="se-75", lamrr=5.87e-03,anlam=5.87e-03,lamsur=5.48e-05 \$end
4	Run CAP88. From the CAP88 output, obtain the air concentration
	value at the receptor location of interest. The CAP88 units will be in pCi/m ³ .
5	Look up the dose factor from Table 2; it will have the units of Ci/m ³
	per 10 mrem per year. Divide the air concentration by this value, then
	multiply by 1 x 10 ⁻¹¹ to obtain the annual dose in mrem.

Federal guidance report method

Federal guidance report method

Radiological dose factors can be obtained from a variety of sources. One of the easiest and quickest sources are the two EPA Federal Guidance Reports (FGRs) available on the web. These contain external and internal dose factors for the original 838 radionuclides evaluated in the ICRP publications. After obtaining exposure rate factors for the four major routes of exposure used by CAP88, they are converted to a Table-2 like dose factor with either a spreadsheet or a computer program. A copy of the 'fgr.pl' program is given as attachment 3. Note that for some cases such as radioactive isotopes of oxygen and neon, both gases, there are no internal dose factors.

This method applies the same equations used by CAP88 to determine the dose contributions from each of the four routes of exposure. The method uses element-specific values provided in the CAP88 file 'allrad88'. If there are no element-specific values available for the radioisotope in question, similar but conservative values taken from 'allrad88' are applied by the 'fgr.pl' program.

Steps to calculate dose

To calculate a dose using the FGRs, perform the following steps:

Step	Action
1	Go to the following web site
	http://homer.hsr.ornl.gov/VLAB/FedGdReps.html and follow the links
	to obtain dose factors for air immersion, contaminated ground surface,
	inhalation, and ingestion. Print out each web page for use in step 2,
	and to be included in the peer review file.
2	The next step is to edit the input file to the 'fgr.pl' Perl5 program. This program is currently run on <i>sibyl</i> . Use the effective dose equivalent value for each route of exposure obtained above, which is labeled h_E on the print-outs. An example of the input program is shown below:
	<pre>"cl-39" => 55.6 minutes "half-life" => 3.8611e-02 #note, this must be in days "immersion" => 7.29e-14 #Sv/Bq per s/m3 "ground-shine" => 1.35e-15 #Sv/Bq per s/m2 "inhalation" => 3.06e-11 #Sv/Bq "ingestion" => 4.96e-11 #Sv/Bq</pre>

Federal guidance report method, continued

Step	Action
3	Once the file has been created and verified with the proper data, run
	the 'fgr.pl' program at the <i>sibyl</i> command prompt, e.g. "fgr.pl cl-39".
	Print the screen output from the program to be included in the peer
	review file. The process should look like the example given below.
	sibyl-kwj>fgr.pl cl-39
	######## Working on EPA Dose Factor Calculations #########
	The element to look for is cl, element cl, not found, using defaults. The B iv1 factor is 1.00e+00
	The B iv2 factor is 1.00e-01
	The F m factor is 1.00e-02
	The F_f factor is 1.00e-01

	The file cl-39.txt is readable, working on "cl-39" => 55.6 minutes
	The lambda factor is 1.80e+01

	The submersion factor is 8.51e+08
	The ground-shine factor is 1.44e+08
	The inhalation factor is 9.09e+07 The ingestion factor is 4.53e+06

	vegetable intake factor is 4.203e+05
	produce intake factor is 4.109e+06
	milk intake factor is 1.614e-09 meat intake factor is 5.643e-149
	meat intake factor is 5.643e-149
	EPA Table 2 equivalent = 9.17e-10
	######### Finished EPA Dose Factor Calculations ##########
4	Now that the Table 2 type dose factor has been calculated, it is possible
	to calculate a dose by following the steps given in the previous chapter
	(Appendix E, Table 2 Method).

Raw data method

Raw data method

This method is required when there are no published or readily obtainable sources of dose conversions for the radionuclide in question. ESH-17 has identified over 25 radionuclides of this type. When this is the case, dose conversions must be derived from the nuclear decay radiation of the particular radionuclide.

The raw data method requires three sets of steps:

- immersion and contaminated ground surface dose factors
- one of the following:
 - inhalation dose factor spreadsheet method
 - inhalation dose factor programs method
- ingestion dose factor programs method

These processes are described in the four sets of steps below. A flow chart outlining the entire process is given as attachment 2.

Raw nuclear decay data is obtained for the radionuclide, then it is reorganized and reformatted by the PERL 5 program 'endsf.pl' for input to the EPA program DOSFACTOR. A copy of the 'endsf.pl' program is included in attachment 3.

Steps to calculate immersion and contaminated ground surface dose factors To calculate immersion and contaminated ground surface dose factors, perform the following steps:

Step	Action
1	Go the Evaluated Nuclear Data Structure File located at the following
	web site:
	http://www.nndc.bnl.gov/nndc/nudat/radform.html
	On the query form, enter the mass number and element symbol for the
	radionuclide of interest. Also, in the 'Table format' drop down menu,
	be sure the –Tabbed file- choice is selected. Download the file to your
	PC.

Step	Action
2	Next, log on to <i>sibyl</i> and FTP the file to your account on <i>sibyl</i> , follow
_	the example below to successfully transfer the file.
	the example below to successfully transfer the file.
	sibyl-kwj>ftp cleanair
	Connected to cleanair.lanl.gov. 220 cleanair Microsoft FTP Service (Version 3.0).
	Name (cleanair:kwj): kwj
	331 Password required for kwj.
	Password: 230 User kwj logged in.
	Remote system type is Windows NT.
	ftp> cd kwjftp
	250 CWD command successful.
	ftp> cd radionuclides 250 CWD command successful.
	ftp> get au-26.tab
	200 PORT command successful.
	150 Opening ASCII mode data connection for al-26.tab(821 bytes). 226 Transfer complete.
	821 bytes received in 0.12 seconds (6.43 Kbytes/s)
	ftp> bye
3	On <i>sibyl</i> , make sure the downloaded tab-file is in the same directory as
	the 'endsf.pl' PERL5 program. The program will read the file, sort out
	the radiations, and write the input file to the DOSFACTOR code. Run
	the program, e.g. "endsf.pl o-13", and print the screen output which
	should be included in the peer review file.
	sibyl-kwj>endsf.pl cl-34m
	The nuclide to be reformatted is cl
	The file cl-34m.tab is readable Energy B-max intensity

	emission B+ 203.4 501.9 0.2640
	emission B+ 554.8 1312.5 25.6000 emission B+ 1099.0 2489.1 28.4000
	emission E 2.1 0.0 1.0170
	emission E 2.4 0.0 5.6600
	emission E 143.5 0.0 6.2400 emission E 146.1 0.0 0.5510
	emission E 146.1 0.0 0.5510 emission G 2.3 0.0 0.0837
	emission G 2.6 0.0 0.5900
	emission G 146.4 0.0 40.5000
	emission G 511.0 0.0 108.5400 emission G 1177.0 0.0 14.0900
	emission G 1573.0 0.0 0.0160
	emission G 1987.0 0.0 0.1850
	emission G 2127.0 0.0 42.8000 emission G 2561.0 0.0 0.0330
	emission G 2749.0 0.0 0.0220
	emission G 3304.0 0.0 12.2900
	emission G 4115.0 0.0 0.2730
	number of alphas = 0
	number of betas = 0
	number of positrons= 3
	number of electrons= 4 number of gammas= 12

	endsf.pl probably ended succesfully
	At this point, you may want to view the output file to see if it was
	successfully created, or move on to the next step.

Step	Action
4	The next step is to run the David C. Kocher's DOSFACTOR code. This is the same program originally used to derive external dose factors for the CAP88 code (EPA, 1989). A small PERL5 script, called 'facdos.pl' is used to run the program. At the <i>sibyl</i> command prompt, type the command 'facdos.pl'. The output from the DOSFACTOR code is the file 'dosout-4'. Print out this file which is to be included in the peer review folder. An example is given below. sibyl-kwj>cat dosout-4 o-15
	4.00E+09 4.33E+09 4.15E+09 4.46E+09 5.98E+09 4.56E+09 3.99E+09 3.57E+09 5.21E+09 1.23E+10 4.06E+09 4.10E+09 5.98E+09 4.53E+09 5.54E+09 3.63E+09 .00E+00 .00E+0
	In this example, the air-immersion EDE factor is 5.10E+09 and has the units of mrem mCi / yr m³. The contaminated ground surface dose factor in this example is 1.05E+06 and has the units of mrem mCi / yr m².
5	Convert these two values by multiplying by the following factor: 8.56435 x 10 ⁻²⁴ . This is a unit conversion and the product will have the same units as the EPA FGR #12 values of Sv Bq ⁻¹ s ⁻¹ m ³ for air submersion and Sv Bq ⁻¹ s ⁻¹ m ² for contaminated ground surface.
6	The numbers converted in step 5 could now be entered into an input file to the 'fgr.pl' program. If the radionuclide is a gas, such as oxygen, nitrogen, or neon, the next step would be to run the 'fgr.pl' program (follow the steps in the previous chapter) to obtain the EPA-Table 2 like dose factor. Otherwise, it is necessary to derive the internal factors as shown in the next section before proceeding with the 'fgr.pl' run.

Inhalation dose factor – spreadsheet method For the inhalation route of exposure, the Air Quality group has developed two separate ways to derive the dose factors. Both methods follow the ICRP-30 lung model. All equations used in this process were taken directly from ICRP-30. The first method, by the use of a spreadsheet, is given in the following steps.

Steps to calculate inhalation dose factor – spreadsheet method To calculate inhalation dose factor by the spreadsheet method, perform the following steps:

Step	Action
1	These steps describe how to generate an inhalation dose factor from
	raw data. First, obtain nuclear decay data for the radioisotope being
	analyzed including half-life, type of decay, energy of decay, and
	frequency of emitted particles per decay. The "Table of Isotopes"
	(Firestone, 1996) or the National Nuclear Data Center at Brookhaven
	National Lab are good sources.
2	The remaining steps deal with calculation contained in an Excel
	spreadsheet. Open the "Inhalation-DCF.xls" spreadsheet, and go to the
	tab named "SEE" and enter the type of decay, energy of decay, and
	number of emitted particles per disintegration in the respective
	columns of the spreadsheet. (The SEE values will be calculated in later
	steps.)
3	Lung Model Go to the tab named "Lung-Model" and verify the
	proper parameters have been entered in table 1. Parameters include the
	deposition fraction (D) for each region. For each of the three
	respiratory retention categories verify the half-time (T) of residence, in
	days, for each compartment and for each fraction (F) leaving the region
	for each class. Also enter the physical half-life and calculate lamda _R
	(in seconds ⁻¹).

Step	Action
4	Activity in Lungs The next step is to apply the equations of ICRP-30,
	PART 1 to calculate the total number of nuclear transformations in the
	lung compartments per unit intake. First, use <i>Table 4</i> to calculate the
	clearance constant (lambda) for each transfer rate (in seconds ⁻¹).
	Second, use <i>Table 3</i> to find the summed transfer rate due to clearance
	and physical decay. Third, use <i>Table 2</i> to calculate the number of
	transformations for each compartment. For most compartments the
	number of transformations is simply the compartment's values of D*F
	divided by the summed lambda. The equations used are taken from
	Table A.1 on page 60 of ICRP-30, PART 1. Assume that the total
	amount of transformations in the lung is the sum of the number of
	transformations per unit intake for compartments c though j
	(transformations in the N-P region are not added in). This is explained
	in the fourth paragraph on page 28 of ICRP-30, PART 1. Repeat this
	step for each clearance category.
5	SEE (Lung←Lung) Next we will calculate the specific effective
	energy (SEE) imparted per gram, with the lung as the target tissue, as a
	consequence of the emissions of each radiation type per nuclear
	transformation in the source (lung). First, go to the "S-to-T" tab in the
	spreadsheet and verify that the correct values of specific absorbed
	fraction (SAF) have been correctly entered into <i>Table 1</i> with the source
	as the lungs and the target as the lungs, for various photon energies
	taken from Table I.2 of ICRP-23. Second, go to the "SEE" tab in the
	spreadsheet and verify the correct values for specific absorbed fraction are entered (column F for lungs); for betas, positrons, and electrons the
	value should be 1.0; for gamma's, the values should match the
	corresponding energy bins given on the "S-to-T" spreadsheet. Third,
	now we can calculate SEE for each radiation type and energy using
	equation 5.6 on page 28 of ICRP-30, PART 1. The quality factor (Q)
	will be 1 in all cases except for alpha particles (use 20). The value of
	SEE is entered in column 5 of the spreadsheet and the sum over all
	types of transformation and energy is provided at the top (and bottom).

Step	Action
6	Lung to Lung Dose At this point it is possible to calculate a dose to the lung with the lung as the source and target. Go to the spreadsheet tab labeled "_dcf_" and verify the correct number of total transformations (U) is entered in column 1 and the correct value for SEE(S←T) is entered in column D. The dose will be:
	Dose = $[U * SEE(S \leftarrow T) * 1.6 \times 10^{-13} \text{ J/MeV} * 1000g/Kg] / (1 J / kg - Gy)$ [1]
	And will have the units of Sv/Bq since we already accounted for Q in step 5. The effective dose factor is calculated by multiplying the result from above by the ICRP-26 organ weighting factor, in this case 0.12, verify the correct value is provided in column 'G' of the "_dcf_" spreadsheet.
7	Activity in GI-Tract To complete the calculation of the dose to the lung, we must calculate the total activity in the other significant source organs, beginning with the gastrointestinal (G.I.) tract. First, go to the spreadsheet tab "GI_Model" and verify that the correct input rates for transfer of activity from the respiratory system has been calculated for each respective retention class, in Table 1. The equation used is found on page 60 of ICRP-30, PART 1. Second, verify that Table 2 contains the correct retention half-times for the various GI tract compartments as given in ICRP-30, PART 1. Also calculated in Table 2 are the constants for retention and the summed constants for retention plus physical decay. Third, verify that in Table 3, the transfer constant from the small intestine to body fluids is correctly calculated, using the f1 value for the radionuclide in question (obtained from NUREG/CR-4884). Finally, calculate the total number of transformations per compartment using the equations given Table A.2 on page 61 of ICRP-30, PART 1.
8	SEE (Lung←GI) The next step is similar to what was done in step 5. We need to determine the specific absorbed energy fractions for each compartment of the GI tract. First, go to the tab "S-to-T" and verify the correct values have been entered from ICRP-23 into Table 3 for the source organ stomach and the target organ lung. Repeat this step for each section of the GI tract (tables 4, 5, and 6). Second, go to the tab "SEE" and verify the correct calculations were performed for each radiation type, radiation energy, frequency, and GI tract source organs, to the lung.
9	GI Dose to Lung First go to the tab "_dcf_" and verify the correct number of transformations for each GI tract compartment as well as the sum of the individual SEE's calculated in step 8 have been correctly entered. Second, verify the proper dose the lung for each of the four GI sources has been calculated (using equation 1 above).

Step	Action
10	Activity in Body Fluids and Receptor Organs The next step is to
	calculate the dose to the lungs resulting from the activity transferred
	from both the respiratory tract and GI tract to body fluids and
	subsequently to other tissues in the body depending upon the metabolic
	model for the element in question. As discussed in ICRP-30, activity is
	input to the transfer compartment 'a' (body fluids and blood) and
	subsequent other receptor tissues from the respiratory tract and GI tract
	according to the equations given on page 18 of PART 1. As a
	conservative simplification to the calculation process, we will assume
	that the receptor tissues will be the 'whole body' or 'other'. If the
	biological half-life is available for the element in question, we will use
	it to calculate the whole body retention constant (lambda _E). The first
	step is to determine the total input to transfer compartment 'a'. Go to the spreadsheet tab "trans_a" and verify the correct retention constants
	are given in Table 1 for each compartment. Second, verify the correct
	data and calculations have been performed in Table 2 to calculate the
	total input from the respiratory tract using the last equation given on
	page 60 of ICRP-30 PART 1. Third, verify the correct data and
	calculations have been performed in Table 3 to calculate the input from
	the GI tract using the second equation from the bottom on page 60 of
	ICRP-30 PART 1. Fourth, verify that the total number of
	transformations has been correctly calculated in Table 4 for both the
	transfer compartment 'a' and the receptor tissues compartment Table 4,
	using the first two equations given at the top of page 58 of ICRP-30
	PART 1. Note, in the case when the biological and physical half-life
	are short relative to the integration period, (T= 50 years) these two
	ICRP-30 equations simplify to the following:
	Ua = [sum of input] / (lambdaA + lambdaR) for transfer compartment 'a' [2]
	Uo = [sum of input * lambdaA]) / (lambdaA + lambdaR) * (lambdaE + lambdaR) for whole body [3]
11	SEE (Lung ← whole body) The next step is to calculate the dose to
	lungs due to activity transferred from the lungs and GI tract to receptor
	organs. For simplification, we will assume the receptor organs to be
	whole body. First go to the tab "S-to-T" and verify that the correct
	values have been entered into table 7 with the whole body as the source
	and the lungs as the target. We will make the conservative assumption
	that the receptor or other tissues will also be the whole body and apply
	those SAFs. At this point, it may be useful to verify the correct entry
	into table 7 for the other target organs with whole body as the source as well.
	WCII.

Step	Action
12	Whole body Dose to Lung First, go to the tab "SEE" and verify the
	correct calculations were performed for each radiation type, radiation
	energy, frequency, and using the SAF's for the whole body and the
	target the lung, these calculations begin in column EF'. Finally, go to
	the "_dcf_" tab and verify the calculations for source in transfer 'a' and
	whole body and dose to lungs.
13	SEE (Lung ← body fluids) The next step is to determine the dose to
	the lungs from due to transformations occurring in the body fluids, or
	as referred to in ICRP-30 Part 1, transfer compartment 'a'. We will use
	the SEE value calculated for whole body as the SEE value for dose to
	the (lung and other target organs) due to activity in transfer
	compartment 'a'. This value should have been verified in step 11.
14	Body Fluids Dose to Lung First, verify the correct value for SEE (as
	described in step 13) has been inserted into the cell labeled 'transf_a' in
	the spreadsheet tab "_dcf_" with the lungs as the target organ. Second
	verify the correct number of transformations has been applied in the
	spreadsheet. These were calculated in the "trans_a" spreadsheet for
	each lung retention class. Finally, ensure the correct dose equivalent
	has been calculated in the "_dcf_" spreadsheet for this source. At this
	point, we have calculated the total dose to the lungs from all of the
	activity initially inhaled (1Bq) and subsequently deposited in the lungs
1.5	and transferred to other tissues within the body.
15	Dose to other ICRP-26 Organs The next step will be to repeat the
	above steps to calculate dose to the other organs for the activity in the
	respiratory tract, GI tract, transfer compartment, and whole body. First, verify the correct values have been entered into the "S_to_T"
	spreadsheet. Second verify the correct calculations in the "SEE"
	spreadsheet. Second verify the correct calculations in the SEE spreadsheet. Third, in the "_dcf_" spreadsheet, verify the correct
	calculations for dose from lungs, stomach, SI, LLI, ULI and whole
	body to the target organs of: gonads, breast, red marrow, bone surfaces,
	and thyroid. Fourth, ensure that the proper organ-weighting factor has
	been applied for each case. For the remainder dose, will we assume the
	following five organs: stomach, small intestine, upper large intestine,
	lower large intestine, and whole body.

Step	Action
16	Remainder Organ Doses As stated above, for simplicity we assumed
	the stomach, small intestine, upper large intestine, lower large intestine,
	and whole body will be the remainder organs. We need to determine
	the specific absorbed energy fractions for each compartment of the GI
	tract. First, go to the tab "S-to-T" and verify the values of <i>Table 2</i> against the equation given for non-penetrating radiation on page 33 of
	ICRP30, PART1:
	$1/2 * 1/ M_c^T * v$
	where M_c^T is the mass of contents of the section of the GI tract, also
	given on page 33, and v is 1 for beta particles, etc. and 0.01 for alpha
	particles. Second, verify that for each non-penetrating (np) radiation
	listed on the "SEE" spreadsheet, the proper value has been entered in
	the correct column for each source/target combination. Third, verify
	the SEE was correctly calculated for both np and penetrating radiation. Fourth, for each GI tract target listed in the "_dcf_" spreadsheet, verify
	that both the correct number of transformations have been entered in
	column F, the SEE value in column D, and the dose in column G.
17	Remainder Whole Body Dose As was stated earlier, we assume the
1	whole body to be one of the remainder organs as a conservative step
	and to simplify the calculations (if sufficient metabolic and biokinetic
	information for the element in question is available, it may be prudent
	to apply other target organs as needed). We will apply SAF value of 1
	for np radiation when whole body is both the source and target. First,
	verify this has been done so in the spreadsheet tab labeled "SEE" and
	that the correct total value for SEE (whole body—whole body) was
	obtained. Second, verify that the correct number of transformations
	and values of SEE were used in the spreadsheet tab "_dcf_". Third,
	verify the correct dose equivalent was calculated in that same spreadsheet.
18	Effective Dose Equivalent Calculation. At this point it is now
10	possible to calculate the effective dose equivalent by applying the
	necessary ICRP-26 organ weighting factors and summing the target
	organ doses for each lung retention class. Go to the spreadsheet tab
	"Table2_dcf" and verify this was done correctly in column 'C' for each
	case.

Step	Action
19	Effective Dose Equivalent and Conversions. The final step is to
	convert the inhalation dose factor into an EPA '40CFR60-Appendix E-
	Table 2' like dose conversion. The EPA Table 2 values are air
	concentrations for a given radionuclide that would result in a 10 mrem
	dose if a person was exposed to that concentration over a period of 1
	year. First, remembering the dose factor was done per Bq of inhalation
	intake, we use the value in column 'C' of the spreadsheet tab "Table
	2_dcf" to make the conversion. Second, we assume a unit air
	concentration and use the default value of breathing rate used by
	CAP88 to arrive at the dose factor given in column 'K' of the
	spreadsheet. Verify the calculation was done correctly. Third, we take
	this value and divide by 10, then take the inverse to arrive at a dose
	conversion factor that is in the same units as those given in EPA's
	Table 2, verify this was done correctly.

Inhalation dose factor – programs method The second method used by ESH-17 to calculate an inhalation dose factor takes advantage of a new program from EPA, named SEECAL (ORNL, 1995). The basic process is the same as the spreadsheet method above; however the target-to-source SEE values are calculated using the SEECAL program. Also, all of the other calculations are accomplished with PERL5 programs developed by ESH-17 and by additional EPA programs that were supplied with the DCAL package. A flow chart that depicts the steps discussed below is shown in attachment 2. This process assumes the user has obtained the DCAL collection of codes, has successfully installed them on the local hard-drive, and has an account on *sibyl* with access to the PERL5 programs.

Steps to calculate inhalation dose factor – programs method To calculate inhalation dose factor by the programs method, perform the following steps:

Step	Action
1	Ensure that the first 3 steps from the chapter "Raw Data Method –
	Immersion and Contaminated Ground Surface Dose Factors" have been
	successfully completed.
2	Run the PERL5 program 'tab2icrp.project leader.' Print the output file
	from the program and include it in the peer review folder.
3	Next, using a large capacity editor such as Ultra-Edit, go to the DCAL
	directory and open the file ICRP38.DAT (this file contains over
	220,000 records). Using the editor, append the output file from the
	'tab2icrp' program to the end of the ICRP38.DAT file. Write down the
	record or line number corresponding to the beginning of the append.

Step	Action
4	While in the DCAL directory, also open the file ICRP38.IDX and
	create a new record for the radionuclide of interest. Enter the
	record/line number from step three in the appropriate location in the
	record. A description of the each record element is given on page 8 of
	the Nucdecay manual that was supplied with the DCAL package. Save
	and close the ICRP38.IDX file once the record has been completed.
5	To ensure the entries made in the ICRP38.DAT and ICRP38.IDX were
	successful and accurate, run the EPA program RADSUM, print out the
	screen data to cross check with the nuclear radiations file, the print-out
	should be included in the peer review folder. Results from the
	RADSUM program are needed to complete the record entries in
	ICRP38.IDX for the radionuclide being analyzed.
6	For the radionuclide being analyzed, it is now possible to run the EPA
	program DEXRAX which will extract the necessary data needed by the
	SEECAL program. Run DEXRAX and print a copy of the output file
	(*.rad) to be included in the peer review folder.
7	Run the SEEPAR program (also part of the DCAL package), for
	Inhalation SEE's use these 'source' organs: ST_Cont, SI_Cont,
	ULI_Cont, LLI_Cont, Lng_Tiss, Blood, and Other. Note, Blood and
	Body_Tis are the same, also Lng_Cont and Lng_Tiss are the same.
	The Target tissues should always be Bone_Sur, Breast, St_Wall,
	SI_Wall, ULI_Wall, LLI_Wall, Lng_Tiss, Muscle, Ovaries,
	R_Marrow, and Thyroid. The radionuclide request file will have the
	extension *.req.
8	It should now be possible to successfully run the SEECAL program,
	follow the screen prompts to complete the execution of the program.
	The output file (*.see) be should be included in the peer review file.
9	Place a copy of the output file from step 8 into the Users's drive under
10	the "FTPfiles' folder.
10	Log on to <i>sibyl</i> , use the FTP utility to transfer the file from the Cleanair
	server to your account on <i>sibyl</i> .

Step	Action
11	The next step will be to generate the metabolic data input file, this file should have the extension of *.bio. A good source of metabolic data for most elements is the 'metadata.dat' file contained in the GENII dose assessment package. Descriptions of the data and format of records contained in the file can be found on page 2.27 of Volume II of the GENII User's Manual. An example input file is shown below.
	17 CHLORINE Adult 2.77 0.95 0.03 0.99 0.99 0.0 Other 1.0 10. 0.95 0.03 2.2222e-02
	In this example 2.77 is the rate constant (d ⁻¹) for the transfer compartment (blood and body fluids), and 0.99, 0.99 and 0.0 are the f1 factors. Also, 'Other' is the target organ and 1.0 is the fraction of material going from the transfer compartment to the target organ, and 10 is the biological half-life [Note: This is incorrectly described in Volume II of the GENII User's Manual.] of the element in the target organ. The last line of the file should be the half-life (in days) of the particular radioisotope being analyzed. The other parameters are not used for now.
12	Using the *.bio and *.see file, it is now possible to generate inhalation dose conversion factors using ESH-17's 'inhdose.pl' PERL5 program. A current copy of the program is given in attachment 3. The program will write out to 11 target organs and calculate an EDE for each of the three lung clearance categories. The output file (*.inh) be should be included in the peer review file. The dose factors will have the units of Sv/Bq intake.
13	The dose factors obtained in step 12 could now be entered into the appropriate record of the input file to the 'fgr.pl' program. It will necessary to derive the ingestion factors as shown in the next section before proceeding with the 'fgr.pl' run.

factor programs method

Ingestion dose The Calculation of Ingestion dose factors also takes advantage of the SEECAL program to calculate source to target SEE values. A PERL5 program performs the final dose calculations using the gastrointestinal tract model of ICRP-30. Currently, ESH-17 does not have this method available in a spreadsheet.

Steps to calculate ingestion dose factor – programs method To calculate ingestion dose factor by the programs method, perform the following steps:

Step	Action			
1	Repeat steps 7 thru 10 of the previous section, however, edit the *.req			
	file to replace the source organ Lng_Tiss with the source organ			
	Body_Tis. The target organs will remain the same.			
2	Using the *.bio and *.see file, it is now possible to generate ingestion			
	dose conversion factors using ESH-17's 'ingdose.pl' PERL5 program.			
	A current copy of the program is given in attachment 3. The output file			
	(*.ing) be should be included in the peer review file. The dose factors			
	will have the units of Sv/Bq intake.			
3	The dose factors obtained in step 2 could now be entered into the			
	appropriate record of the input file to the 'fgr.pl' program. Provided			
	that the immersion, ground-shine, inhalation and ingestion dose factors			
	have been entered correctly it is now possible to run the 'fgr.pl'			
	program. Refer to the chapter Federal Guidance Report Method for			
	instructions.			

Checks and peer review

and peer review of dose factor data

Records check Because of the significant amount of data and calculations used in deriving dose factors for non-CAP88 radionuclides, obtain a peer review of the data for each radioisotope used in the process. Document the peer's review on the forms that were reviewed.

> Also, as an added insurance to the accuracy and reliability of the calculations, run test cases. Keep a folder containing all of the data used and the records specified in each of the above chapters.

> For the "Raw Data Method" only, complete the two checklists (attachments 4 and 5) to document the completeness and review of the data.

Maintain table of dose factors

As new factors are derived, update the spreadsheet table of all the non-CAP88 dose factors that have been generated by ESH-17. This table will be used as discussed in procedure ESH-17-511 ("Calculating mrem per Ci Factors").

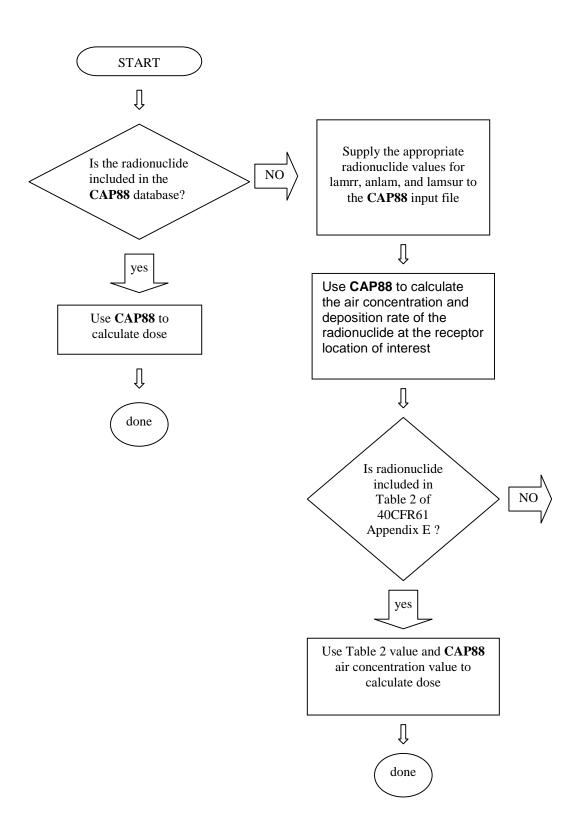
Records resulting from this procedure

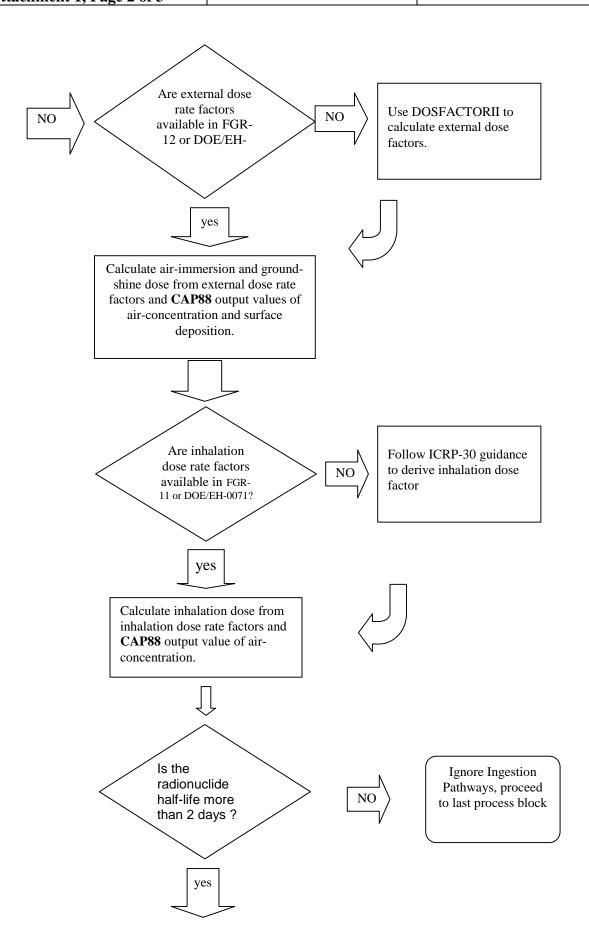
Records

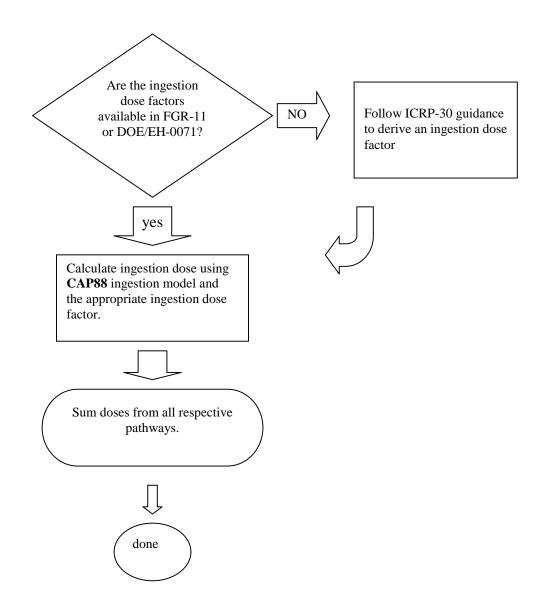
The following records generated as a result of this procedure are to be submitted **within 2 weeks** as records to the records coordinator:

- folder for each non-CAP88 radionuclide, containing all data and records specified in the chapter for the method used
- updated list of dose factors for non-CAP88 radionuclides
- Check Sheet for Dose Factors from Nuclear Decay Data (attachment 4) (when raw data method used)
- Quality Assurance Review of Dose Factors Calculated from Nuclear Decay Data (attachment 5) (when raw data method used)

FLOW CHART OF PROCESS TO CALCULATE NON-CAP88 RADIONUCLIDE DOSES

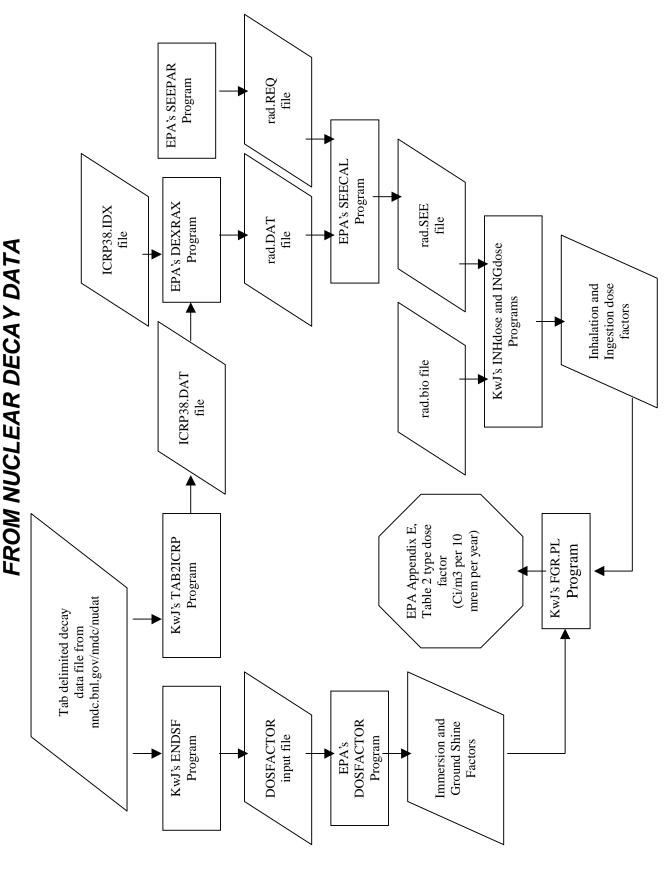






ESH-17-512, R0 Attachment 2, Page 1 of 1

FLOW CHART OF STEPS TO CALCULATE NON-CAP88 RADIONUCLIDE DOSES



PERL 5 PROGRAMS USED

```
fgr.pl
#!/usr/bin/perl
#06-Jan-00==last update KwJ
# Perl program to convert EPA's FGR values
# to EPA Appendix E, table 2 dose factor.
die "\t Needed radionuclides on command line!\n" unless @ARGV;
print "######## Working on EPA Dose Factor Calculations ########\n";
####################################
# read allrad88 values for element
# specific transfer factors.
$B iv1=1.0 ; $B iv2=1.0e-1 ; $F m=1.0e-2 ; $F f=1.0e-1 ;
while (@ARGV) {
$next = shift @ARGV;
$file in = $next . ".txt";
# assign the first letters of the command line to $1
if ( pext = m/([a-z]*)/x ) { pext = m/([a-z]*)/x )
    print " The element to look for is $name,";
open(TRANSFERS_FILE, "<element_data") || die "element_data not opened";
while (<TRANSFERS FILE>)
    if ( m/ ^$name /x )
     { print " element $name, found in the transfers file.\n";
         (\nc, \c) = \nc, \c)
         close (TRANSFERS FILE);
         $found = "true" ;
if(!$found) { print "\t element $name, not found, using defaults.\n"; }
printf("%30s %-7.2e\n", "The B_iv1 factor is",$B_iv1);
printf("%30s %-7.2e\n", "The B_iv2 factor is",$B iv2);
printf("%30s %-7.2e\n", "The F_m factor is",$F_m);
printf("%30s %-7.2e\n", "The F f factor is",$F f);
print "
                       # initialize the rest of the parameters
$breathing rate = 917000;
$vd=0.18;
$wetd=0.01;
\frac{1}{2} $lamsur = 5.48e-05;
#initialize the ingestion paramters #
R_veg = 0.2;
R = 0.57;
t = veg = 1440;
t_e_grass = 720;
t = feed = 720;
$Y v = 0.716;
Y v grass = 0.28;
$t b = 15.0 * 365.25 *24;
P = 2.15e+2;
t h = 340;
t_h_grass = 0;
t_h_f = 90*24;
$U v = 18;
$U p = 176;
garden f = 1;
home_f = 1;
```

```
\$wash off = 0.5;
# milk and meat only parameters
f p = 0.4;
fs = 0.43;
Q f = 15.6;
$t f = 2;
$U m = 112;
$t s = 20;
$U_f = 85;
open(INFILE, $file_in) || die "file INFILE not opened";
if (-r INFILE) {print " The file $file_in is readable,"; }
@input = <INFILE>;
close (INFILE);
chomp @input;
print "\ working on $input[0]\n";
foreach $line (@input)
  ($label,$div,$value) = split('', $line);
  if ($label =~ /half-life/) { $lambda = &lambda_calc; }
if ($label =~ /immersion/) { $rtel = &immersion calc:
  if ($label =~ /immersion/)
                                 { $rte1 = &immersion_calc; }
  if ($label =~ /ground-shine/) { $rte2 = &shine_calc; }
  if ($label =~ /inhalation/) { $rte3 = &inhalation_calc; }
if ($label =~ /ingestion/) { $rte4 = &ingestion_calc; }
print "
           **********************************
printf("%30s %-7.2e\n", "The lambda factor is",$lambda);
print " *******************************
n";
printf("%30s %-7.2e\n", "The submersion factor is",\$rte1); printf("%30s %-7.2e\n", "The ground-shine factor is",\$rte2);
printf("%30s %-7.2e\n", "The inhalation factor is",$rte3 );
printf("%30s %-7.2e\n", "The ingestion factor is",$rte4);
         printf("%30s %-8.3e\n", " vegetable intake factor is",$leafy );
printf("%30s %-8.3e\n", " produce intake factor is",prodc);
printf("%30s %-8.3e\n", " milk intake factor is",$milk );
printf("%30s %-8.3e\n", " meat intake factor is",$meat );
$all rtes= $rte1 + $rte2 + $rte3 + $rte4;
$comply factor = (1.0/$all rtes);
#
           print "
printf("%30s %7.2e\n", "EPA Table 2 equivalent = ",$comply factor);
print "######### Finished EPA Dose Factor Calculations ########\n";
 #End main loop after all arguments on command line are read in.
##### end of main program #####
sub lambda calc
{0.693147/$value }
sub immersion calc
{ ($value *3.70e+14 *365.25 *24 *3600) }
sub shine calc
{ ($value * 3.70e+12 *365.25 *24 *3600) *
   (1-exp(-($lambda+$lamsur)* 365.25))/($lambda+$lamsur) *
   ($vd+$wetd) *86400 }
sub inhalation calc
 { $value * 3.7e+08 * $breathing rate * 8760 }
sub ingestion calc
 {\$lamhr = \$lambda/24;}
# leafy vegetable ingestion dose calcualtion
  $leafy =
   $value * 3.7e+14 *
     ($vd+$wetd) * 3600 * 0.01 *
```

```
($R_veg *(1-exp(-($lamw+$lamhr)*$t_e_veg))) / ($Y_v *($lamw+$lamhr))
      + ($B iv2 *(1-exp(-$lamhr*$t b))) / ($P * $lamhr)
      * (exp(-$lamhr*$t_h))
      * $U v * $garden f * $home f * $wash off;
# produce is just the ratio of annual intakes
        prodc = product | product 
# milk dose calc
    Cp =
        ($vd+$wetd) *3600 *0.01 *
        ($R grass *
         (1-exp(-(\$lamw+\$lamhr)*\$t_e\_grass))) \ / \ (\$Y_v\_grass *(\$lamw+\$lamhr)) 
      +($B iv1*(1-exp(-$lamhr*$t b)))/($P*$lamhr)*(exp(-$lamhr*$t h grass))
      *($P * $lamhr) *(exp(-$lamhr*$t_h_grass))
      );
    $C s =
        ($vd+$wetd) *3600 *0.01 *
        ($R grass *(1-exp(-($lamw+$lamhr)*$t e grass))) /
        ($Y v grass *($lamw+$lamhr)) + ($B iv1 *(1-exp(-$lamhr*$t b))) /
        ($P * $lamhr) *(exp(-$lamhr*$t h feed))
       );
# activity on grass and feed
   C_v = C_p*(f_p*f_s) + C_s*(1-f_p) + C_s*f_p*(1-f_s);
# dose factor milk pathway
        milk = C_v * F_m * Q_f * exp(slambda * -st_f) *
        $U m * 3.7e+14 * $value;
# dose factor meat pathway
        meat = C_v * F_f * Q_f * exp(lambda * - t_s) *
        $U f * 3.7e + 14 * $value;
# sum each ingestion route
 $sum = $leafy + $prodc + $milk + $meat;
return $sum };
##### end of subroutines #####
endsf.pl
#!/usr/bin/perl
# endsf.pl
# 25-Feb-00==last update KwJ
# Perl program to convert ENDSF tab file to an input
# file for Dr. David Kocher's DOSFACTOR program.
# The default assumption is that the number of quanta
# must be reduced to the allowed number in DOSFACTOR.
# input energies are in keV, output units are MeV
sub descend { $b <=> $a; }
sub ascend { $a <=> $b; }
die "\t Needed radionuclide(s) on command line!\n" unless @ARGV;
while (@ARGV) {
$radin = shift @ARGV;
$ENDSF IN = $radin . ".tab";
# assign the first letters of the command line to $1
print " The nuclide to be reformatted is $rad \n" ; }
open(INFILE, $ENDSF_IN) | | die "Died...file $ENDSF_IN not opened";
if (-r INFILE) {print " The file $ENDSF_IN is readable\n"; }
```

```
@endsf table = <INFILE>;
close (INFILE);
($header) = @endsf table ;
if($header =~ m/Element/ ) { shift @endsf_table ; }
chomp @endsf table;
$no alpha=0;
$no beta=0;
$no posit=0;
$no_elec=0;
$no gamma=0;
                      energy B-max intensity \n";
print "
print "**********************************
n";
foreach $line (@endsf table)
 ($A,$element,$Z,$decay mode,$half,$half txt,
 $radiation,$mod,
 $energy,$endpt,$intensity,$dose) = split( /\t/ , $line);
# Note the energy, endpt, and intensity scalars MAY consit of a
# two word string, the 2nd word should be rejected for now.
($energy, $uncert) = split / /, $energy;
($endpt,$uncert) = split/ /, $endpt;
($intensity, $uncert) = split/ /, $intensity;
# This module shows the data, sorts into arrays based on radiation
# type, and applies conversion needed for DOSFACTOR units.
unless ($mod =~ /TOT/ )
  printf("%10s %-2s %7.1f %7.1f %8.4f \n",
   "emission ", $radiation ,$energy, $endpt, $intensity);
   ext{senergy} = ext{senergy}/1000.00;
   $endpt = $endpt/1000.00 ;
   $intensity = $intensity/100.00;
   if(\$radiation =~ /A/) {
     $no alpha++ ; push @alpha list , ("$intensity $energy") ;
   elsif (\$radiation =~ /B\setminus-/) {
     $no_beta++ ; push @beta_list , ("$intensity $energy $endpt") ;
   elsif (\$radiation =~ /B\+/) {
     $no_posit++ ; push @posit_list , ("$intensity $energy $endpt");
   elsif($radiation =~ /E/) {
      $no elec++ ; push @elec list , ( "$intensity $energy" ) ;
   elsif($radiation =~ /G/ ) {
     $no gamma++ ; push @gamma list , ( "$intensity $energy" ) ;
  else {
     print "**** the radiation type was not recognized***n" ;
  } # ends unless block
 } # ends foreach loop
print "***********************************
n";
print "number of alphas = $no_alpha \n";
print "number of betas =
                            $no_beta \n";
print "number of positrons= $no_posit \n";
print "number of electrons= $no_elec \n";
print "number of gammas=
                           $no gamma \n";
# here is where we sort out the least frequent radiations
if ($no_alpha > 20 )
```

```
{ $extra=$no alpha-20 ;
  print " *** Must sort out $extra Alphas***\n";
@alpha_list = (sort descend @alpha_list);
while ($no_alpha > 20 )
pop @alpha_list ;
$no alpha -- ;
#
#
if ($no_beta > 50 )
 { $extra=$no_beta-50 ;
  print " *** Must sort out $extra Betas***\n";
@beta_list = (sort descend @beta_list);
while ($no_beta > 50 )
{
pop @beta_list ;
 $no beta -- ;
if ($no posit > 20)
{ $extra=$no_posit-20 ;
  print " *** Must sort out $extra Positrons***\n";
@posit list = (sort descend @posit list);
while ($no posit > 20 )
pop @posit_list ;
 $no_posit -- ;
if ($no_elec > 125 )
  print " *** Must sort out $extra Electrons***\n";
@elec_list = (sort descend @elec_list);
while ($no_elec > 125 )
pop @elec_list ;
$no elec -- ;
if ($no gamma > 200 )
  { $extra=$no_gamma - 200 ;
   print " *** Must sort out $extra Gammas***\n";
@gamma_list = (sort descend @gamma_list);
while ($no gamma > 200 )
pop @gamma_list ;
 $no gamma -- ;
#### output to DOSFACTOR input file begins here ####
open(DOSFACin, ">endsf.out");
print DOSFACin sprintf ("%-8s %12s
                                      %5d \n",
      $radin, lc($half txt), $Z);
@alpha_list = &aswap(@alpha_list);
print \overline{\text{DOSFACin}} sprintf("%4d\sqrt{n}",$no alpha);
while( $no_alpha > 0 )
 for (\$i=0 ; \$i <= 2 ; \$i++)
```

```
$line = shift @alpha_list;
    ( $energy , $intensity ) = split ' ' , $line ;
    print DOSFACin sprintf ("%11.4e%11.4e",
           $energy,$intensity);
  }
  print DOSFACin "\n";
   no alpha = no alpha - 3
} # end the alphas loop
@beta_list = &bswap(@beta_list);
print DOSFACin sprintf("%4d\n",$no_beta);
while($no beta > 0 )
{
  for (\$i=0 ; \$i <= 1 ; \$i++)
   {
    $line = shift @beta list;
    ( $endpt , $energy , $intensity ) = split( ' ' , $line) ;
    print DOSFACin sprintf ("%11.4e%11.4e%11.4e",
          $endpt,$energy,$intensity);
  print DOSFACin "\n";
  $no beta = $no beta - 2
 } # end the betas loop
@posit_list = &bswap(@posit_list);
print DOSFACin sprintf("%4d\n",$no posit);
while($no posit > 0)
 for (\$i=0 ; \$i <= 1 ; \$i++)
    $line = shift @posit list;
     ($endpt,$energy,$intensity) = split('', $line);
    print DOSFACin sprintf ("%11.4e%11.4e%11.4e",
          $endpt,$energy,$intensity);
  print DOSFACin "\n";
   $no_posit = $no_posit - 2
 } # end the posits loop
#
#
#foreach (@elec_list) {print "elec= $_ \n" ; }
@elec_list = &aswap(@elec list);
print DOSFACin sprintf("%4d\n",$no elec);
while($no elec > 0)
 {
 for (\$i=0 ; \$i <= 2 ; \$i++)
    $line = shift @elec list;
     ($energy,$intensity) = split('', $line);
    print DOSFACin sprintf ("%11.4e%11.4e",
           $energy,$intensity);
  print DOSFACin "\n";
  $no elec = $no elec - 3
 } # end the Electrons loop
#foreach (@gamma_list) {print "gamma= $_ \n" ; }
@gamma list = &aswap(@gamma list);
print DOSFACin sprintf("%4d\n",$no gamma);
while($no_gamma > 0)
 for (\$i=0 ; \$i <= 2 ; \$i++)
```

```
$gline = shift @gamma_list;
    ($energy,$intensity) = split('', $gline);
    print DOSFACin sprintf ("%11.4e%11.4e",
          $energy,$intensity);
  }
  print DOSFACin "\n";
  $no gamma = $no gamma - 3
} # end the Gammas loop
close (DOSFACin);
} #End main loop after all arguements on command line are read in.
unless( $! = 0 ) {print "***endsf.pl probably ended succesfully***\n"};
sub aswap {
reset '2';
foreach $line(@ )
($intensity, $energy) = split ' ', $line;
push @2 , ("$energy $intensity");
my @swapa = (sort ascend @2) ;
return @swapa ;
*************************************
sub bswap {
reset '3';
foreach $line(@ )
($intensity,$energy,$endpt)
   = split(' ' , $line);
push @3 , ("$endpt $energy $intensity");
  @swapb = (sort ascend @3 ) ;
return @swapb ;
# end of subroutines
facdos.pl
#!/usr/bin/perl
# facdos.pl
# 26-Jan-00==last update KwJ
# Perl program to run DOSFACTOR
system ("cp endsf.out fort.5");
system ("cp endsf.out dosdat");
system ("dosfac1.x < fort.5");</pre>
system (catty);
system ("dosfac4.x < dosdat");</pre>
system ("mv dosout dosout-4");
system ("clear");
print "\n\n";
print "Do you want to delete the ftn.. files (y/n)? ";
$answ = <STDIN>; chomp $answ ;
if ($answ eq "y") { system("rm ftn*");
                  print " ok they are gone \n" ; }
elsif ($answ eq "n") { print " -ok they are still here \n" ; }
sibyl-kwj>
```

```
inhdose.pl
#!/usr/bin/perl
# inhdose.pl
# 23-Feb-00==last update KwJ
# ICRP-30 lung model calculations
# reads SEE values from file radin.see,
# reads metabolic data from file radin.bio
# writes dose factors to file radin.inh
sub descend { $b <=> $a };
die "\t Needed radionuclide(s) on command line!\n" unless @ARGV;
while (@ARGV) {
$radin = shift @ARGV;
# the *.see file contains see's for 6 sources & 11 targets
$SEE IN = $radin . "ih.see";
####-get the first letters of the command line from $1-####
if ( \frac{1}{x} = \frac{1}{x}  ( \frac{1}{x} = \frac{1}{x} ) \frac{1}{x} = \frac{1}{x}
 print " -the inhalation dose factor is for $rad-\n" ; }
open(SEEFILE, $SEE_IN) | | die "Died...file $SEE_IN not opened";
if (-r SEEFILE) {print " -the file $SEE IN is readable-\n"; }
open(OUTPUT, ">$radin.inh");
print OUTPUT
"\$#Equivalent dose (Sv) to target organs per Bq intake-inhalation\n";
# the *.bio file contains the metabolic parameters for the models
# only lamTR, f1, lamTI, biol.,phys.-half-life are being used
# for now, the rest of the parameters are ignored
open(BIOFILE, "<$radin.bio") || die "file $radin.bio not opened\n";
if (-r BIOFILE) {print " -the file $radin.bio is readable-\n"; }
@bio data=<BIOFILE>;
close (BIOFILE);
chomp @bio data; shift @bio data;
($name,$body,$lamTR,$F_U) = split(' ', @bio_data[0]);
($fld,$flw,$fly) = split(' ', @bio_data[1]);
@fh = (sort descend (\$fld, \$flw, \$fly));
$f1 = @fh[0]; # use the highest value for now
print "**** f1 = $f1 ****\n";
($t organ,$F_TI,$bhalfl,$F_OU) = split(' ', @bio_data[2]);
$phalfl = @bio data[3];
if( $phalfl =~ m/End/)
 die "***you need to replace the word 'End' in the $radin.bio file
 with the half-life (in days) for $radin***\n";
######-initial data-####
@class[0] = "D";
@class[1] = "W";
@class[2] = "Y";
n = 1.58e-09; # number of seconds in 50 years
1n2 = log(2);
$s d = 60*60*24;
na = 1e-09 ;
$infinity = 1e+09;
\frac{1}{2} $\lambda = \lambda \ln2/(\lambda \text{phalfl * \lambda s_d}) ;
```

```
#####-organ weight hash--####
"SI_Wall" , 0.06 , "ULI_Wall" , 0.06 ,
                       "LLI_Wall" , 0.06 ,
                       "Lng_Tiss" , 0.12 ,
                                            , 0.06 ,
                       "Muscle"
                       "Ovaries"
                                             , 0.25 ,
                       "R_Marrow" , 0.12 , "Thyroid" , 0.03 ) ;
#####-lung parameters-####
$Dnp = 0.30;
Dtb = 0.08;
p = 0.25;
@a htime = (0.01, 0.01, 0.01); @F a = (0.5, 0.1, 0.01);
@b_htime = (0.01, 0.40, 0.40); @F_b = (0.5, 0.9, 0.99);
@c_htime = (0.01, 0.01, 0.01) ; @F_c = (0.95, 0.5, 0.01) ;
@d_htime = (0.20, 0.20, 0.20) ; @F_d = (0.05, 0.5, 0.99) ;
@f_htime = ( $na, 1.0, 1.0) ; @F_f = ($na, 0.4, 0.4) ;
@j_htime = ( $na, $na, $infinity);@F_j = ($na, $na, 0.1) ;
#####-GI parameters-####
    $lamST = 1 / (1/24 * $lamSI = 1 / (4/24 * )
                                                    $s d);
                                                    $s_d);
    \frac{1}{24} *
                                                   $s d);
    \frac{1}{24}
                                                 $s_d);
    $lamBF = ($f1 * $lamSI) / (1-$f1) ;
#####-tranfer compartment(tr) and receptor tissues(ti) parameters-#####
    $lamTR = $lamTR/$s d;
    \frac{1}{2} $\lambda \lambda \l
while (<SEEFILE>)
{
  unless (! m/TARGETS RAD/)
    print "***** Input data located in $SEE IN input file. *****\n";
    while(<SEEFILE>) { push @sees , $ ; }
close (SEEFILE);
chomp @sees;
$header = shift @sees;
header =  s/-//g;
=  s/\.//g;
push @srcorg , (split ' ', $header);
foreach $rec(@sees) ###main calculational loop begins here###
    (\text{strgt org}, \text{st}, \text{@see}[0], \text{@see}[1], \text{@see}[2], \text{@see}[3], \text{@see}[4], \text{@see}[5]) =
        (split ' ' , $rec);
       print "$trgt org,@see[0],@see[1],@see[2],@see[3],@see[4],@see[5] \n";
print OUTPUT sprintf("%-9s",$trgt org);
for (\$k=0; \$k < 3; \$k++) \# loop for each clearance category
  {
    \frac{1}{2} = \frac{1}{2} (@a_h time [$k] * s_d) ;
```

```
\frac{1}{2} $\text{lamB} = \frac{1}{2} (@b \text{htime}[\frac{1}{2}k] *\frac{1}{2}s \, d) ;
  \frac{1}{2} $\lambda = \frac{1}{2} \left(\text{@c_htime}[\frac{1}{2}k] *\frac{1}{2}s_d\right)
  \frac{1}{2} $\text{lamD} = \frac{1}{2} \left( \text{@d htime} \text{[$k] * $\frac{1}{2}s$ d)
  led{slamE} = ln2/(@e_htime[$k]*$s_d)
  \frac{1}{2} = \frac{1}{2} (ef_htime[$k] * s_d) ;
  \frac{1}{2} = \frac{1}{2} (@g_htime[$k] * s_d) ;
  \frac{1}{2} $\lambda = \frac{1}{2}(\text{@h htime}[\frac{1}{2}k] *\frac{1}{2}s d) ;
  \frac{1}{2} = \frac{1}{2} (\text{@i htime} [\text{$k}] * \text{$s d}) ;
  \frac{1}{2} = \frac{102}{(@j_htime[$k] * s_d)};
#####-number of disintegrations in lung compartments-####
#
  a = (Dnp*@F a[$k])/(amR+$lamA);
  $b_dis=($Dnp*@F_b[$k])/($lamR+$lamB);
  $c_dis=($Dtb*@F_c[$k])/($lamR+$lamC);
  d_dis= (\beta Dtb*@F_d[\k])/(\beta IamR+\beta IamD) + \beta Dp/(\beta IamD+\beta IamR) *
    ((\$lamF*@F f[\$k])/(\$lamF+\$lamR))+((\$lamG*@F g[\$k])/(\$lamG+\$lamR));
  e = \frac{p^*@F e[$k]}{3}
  f_dis=(p_*@F_f[$k])/(f_amR+f_amF);
#
  g_dis=(p_*@F_g[$k])/(s_amR+s_amG);
#
  $h_dis=($Dp*@F_h[$k])/($lamR+$lamH);
  $i dis=($Dp*@F h[$k]*$lamH*@F i[$k]) / (($lamH+$lamR)*($lamI+$lamR));
  j_dis=(p^*@F_h[$k] * lamH*@F_j[$k]) * (1-exp(-$n*$lamR)) /
          (($lamH+$lamR)*$lamR) ;
  $lung dis=$c dis+$d dis+$e dis+$f dis +
                     $g_dis+$h_dis+$i_dis+$j ;
# printf("%s%8.1f\n","transformations in compartment a...",$a_dis);
# printf("%s%8.1f\n","transformations in compartment b...",$b dis);
# printf("%s%8.1f\n","transformations in compartment c...",$c dis);
# printf("%s%8.1f\n","transformations in compartment d...",$d_dis);
# printf("%s%8.1f\n","transformations in compartment e...",$e dis);
# printf("%s%8.1f\n","transformations in compartment f...",$f_dis);
# printf("%s%8.1f\n","transformations in compartment g...",$g dis);
# printf("%s%8.1f\n","transformations in compartment h...",$h dis);
# printf("%s%8.1f\n","transformations in compartment i...",$i_dis);
# printf("%s%8.1f\n","transformations in compartment j...",$j dis);
# printf("%s%8.1f\n","total transformations in lung.....",
    $lung dis);
####-lung activity transfered to the GI tract(gi_in)-#####
  ($Dnp*@F b[$k]*$lamB)/($lamB+$lamR)+
   (\$Dtb*@F d[\$k]*\$lamD)/(\$lamD+\$lamR) + \$Dp *
  ((@F f[$k] * $lamF) / ($lamF + $lamR) + (@F g[$k] * $lamG) / ($lamG + $lamR)) *
   $lamD/($lamD+$lamR);
#
\# printf("%s%6.2e\n","lung activity to gi...",$gi_in );
#####-number of disintegrations in GI compartments-####
$ST_dis = $gi_in / ($lamST + $lamR);
$SI_dis = ($gi_in * $lamST) /(($lamST+$lamR)*($lamSI+$lamBF+$lamR));
$ULI_dis = ($gi_in * $lamST * $lamSI) /
```

```
(($lamST+$lamR) * ($lamSI+$lamBF+$lamR) * ($lamULI+$lamR));
$LLI dis = ($gi in * $lamST * $lamSI * $lamULI) /
          ((\$lamST + \$lamR) * (\$lamSI + \$lamBF + \$lamR) * (\$lamULI + \$lamR) *
          ($lamLLI+$lamR));
# printf("%s%8.1f\n","transformations in stomach.....",$ST dis);
# printf("%s%8.1f\n","transformations in small Int...",$SI dis);
# printf("%s%8.1f\n","transformations in U.L. Int....",$ULI_dis);
# printf("%s%8.1f\n","transformations in L.L. Int....",$LLT dis);
######lung activity transported to the transfer compartment(ln_tr)-#####
  $ln tr=
  ($Dnp*@F_a[$k]*$lamA)/($lamA+$lamR)+
   (\$Dtb*@F_c[\$k]*\$lamC)/(\$lamC+\$lamR) + \$Dp * 
  (@F_e[$k] *$lamE) / ($lamE + $lamR) + $Dp *
  (@F\ h[\$k] *\$lamH*@F\ i[\$k] *\$lamI)/(\$lamH+\$lamR) *(\$lamI+\$lamR);
\# printf("%s%6.2e\n","lung activity to tr...",$ln_tr);
#####-GI activity transported to the transfer compartment(qi tr)-#####
 $gi tr=($lamST*$lamBF)/(($lamST+$lamR)*($lamSI+$lamBF+$lamR));
# printf("%s%6.2e\n", "GI activity to tr.....",$gi tr );
#####-number of disintegrations in tr and ti compartments-####
 tr_dis = (\ln_tr+\sin_t)/(\sin_tr+\sin_t);
# printf("%s%8.0f\n","tr.....,$tr dis );
 $ti dis = ($F TI * $lamTR * ($ln tr+$gi tr)) /
             (($lamTR+$lamR) * ($lamTI+$lamR));
# printf("%s%8.0f\n","ti.....",$ti dis );
####-end of the disintegrations calculations-####
#####-calculate doses and print out results here-#####
   printf("%24s%6.2e\n","Dose to $trgt_org = ",&ORGANDOSE );
   print OUTPUT sprintf(" %1s %7.3e",@class[$k],&ORGANDOSE);
   @effec[$k] = @effec[$k] + &ORGANDOSE * $org wt{$trgt org} ;
  } #end of clearence loop
   print OUTPUT "\n";
 } #end of main calculational loop
   print OUTPUT sprintf("%9s %1s %6.3e %1s %6.3e %1s %6.3e","Effective",
       @class[0],@effec[0],@class[1],@effec[1],@class[2],@effec[2]);
   print OUTPUT "\n-----\n";
} # end of main input loop
close(OUTPUT);
unless( $! = 0 ) {print "***inhdose.pl ended succesfully***\n"};
sub ORGANDOSE {
  $current_dose = $ST_dis * @see[0] +
                  $ULI_dis * @see[2] +
                  $LLI_dis * @see[3] +
                  $lung dis *
                              @see[4] +
                  $tr dis * @see[5] +
                  $ti dis
                          * @see[5] ;
 return $current_dose; }
```

```
ingdose.pl
_ _ _ _ _ _ _ _ _ _ _ _ _ _
#!/usr/bin/perl
# ingdose.pl
# 01-Mar-00==last update KwJ
# ICRP-30 GI model calculations
# reads SEE values from file rad.see,
# reads metabolic data from file rad.bio
# writes dose factors to file rad.ing
# lamTR = rate constant for material leaving the transfer compartment
# lamTI = rate constant for matarial leaving other or target tissues
sub descend { $b <=> $a };
die "\t Needed radionuclide(s) on command line!\n" unless @ARGV;
while (@ARGV) {
$radin = shift @ARGV;
# the rad.see file contains see's for 6 sources & 11 targets
$SEE IN = $radin . "ig.see";
####-get the first letters of the command line from $1-####
if ( \frac{1}{x} = \frac{m}{([a-z]^*)} / x ) { \frac{1}{x} = \frac{1}{x} = \frac{1}{x}
 print " -the ingestion dose factor is for $rad-\n" ; }
open(SEEFILE, $SEE_IN) || die "Died...file $SEE_IN not opened";
if (-r SEEFILE) {print " -the file $SEE IN is readable-\n"; }
open(OUTPUT, ">$radin.ing");
print OUTPUT
"\$#Equivalent dose (Sv) to target organs per Bq intake-ingestion\n";
# the *.bio file contains the metabolic parameters for the models
# only lamTR, f1, biologcial, and physical-half-life are being
# used for now, the rest of the parameters are ignored
open(BIOFILE, "<$radin.bio") || die "file $radin.bio not opened\n";
if (-r BIOFILE) {print " -the file $radin.bio is readable-\n"; }
@bio data=<BIOFILE>;
close (BIOFILE);
chomp @bio data; shift @bio data;
($name,$body,$lamTR,$F U) = split(' ', @bio data[0]);
($fld,$flw,$fly) = split(' ', @bio_data[1]);
@fh = (sort descend ($fld, $flw, $fly));
$f1 = @fh[0]; # use the highest value for now
#if ( $f1 = 1.0 ) { $f1 = 0.99 ; }
($t organ, $F TI, $bhalfl, $F OU) = split(' ', @bio data[2]);
$phalfl = @bio_data[3];
if( $phalfl =~ m/End/)
 die "***you need to replace the word 'End' in the $radin.bio file
 with the half-life (in days) for $radin***\n";
######-initial data-####
n = 1.58e-09; # number of seconds in 50 years
1n2 = log(2);
$s d = 60*60*24 ;
\sin = 1e-09;
$infinity = 1e+09;
\frac{1}{2} $\lambda = \frac{1}{2}(\frac{1}{2}\text{phalfl} * \frac{1}{2}\text{s} d) ;
#####-organ wt hash--####
"St_Wall" , 0.06 ,
```

```
, 0.06 ,
                         "SI Wall"
                                                 , 0.06 ,
                         "ULI_Wall"
                         "LLI Wall"
                                                 , 0.06 ,
                         "Lng_Tiss"
                                                , 0.12 ,
                         "Muscle"
                                                 , 0.06 ,
                                                 , 0.25 ,
                         "Ovaries"
                         "R_Marrow" , 0.12 ,
                         "Thyroid"
                                                  , 0.03);
#####-GI parameters-####
    \frac{1}{24} = \frac{1}{124} + \frac{1}{24}
    \frac{1}{2} = 1 / (4/24 * \frac{1}{2} - \frac{1}{2});
    \frac{1}{24} * \frac{1}{24} * \frac{1}{24} 
    \frac{1}{24} = 1 / (24/24 * \frac{1}{24} ;
    $lamBF = ($f1 * $lamSI) / (1-$f1) ;
#####-tranfer compartment(tr) and receptor tissues(ti) parameters-#####
    \frac{1}{2} $\text{lamTR} = \text{$lamTR}/\text{$s d};
    \frac{1}{2} $\lambda \lambda \l
#
while (<SEEFILE>)
 unless ( ! m/TARGETS RAD/ )
   print "**** Input data located in $SEE IN input file. ****\n";
    while(<SEEFILE>) { push @sees , $_ ; }
close (SEEFILE);
chomp @sees;
$header = shift @sees;
header =  s/-//g;
= s/\.//g;
push @srcorg , (split ' ', $header);
foreach $rec(@sees) ###main calculational loop begins here###
{
    (split ' ' , $rec);
       print "$trgt org,@see[0],@see[1],@see[2],@see[3],@see[4],@see[5] \n";
#
print OUTPUT sprintf("%-9s",$trgt org);
#for (\$k=0; \$k < 3; \$k++) # calculate for each clearance category
  {
#
    $gi_in = 1 ; # 1 Bq intake
######activity in GI compartments#####
ST dis = Sgi in / (SlamST + SlamR);
$SI_dis = ($gi_in * $lamST) /(($lamST+$lamR)*($lamSI+$lamBF+$lamR));
$ULI_dis = ($gi_in * $lamST * $lamSI) /
                       ((\$lamST+\$lamR)*(\$lamSI+\$lamBF+\$lamR)*(\$lamULI+\$lamR));
$LLI dis = ($gi in * $lamST * $lamSI * $lamULI) /
                       (($lamST+$lamR) * ($lamSI+$lamBF+$lamR) * ($lamULI+$lamR) *
                       ($lamLLI+$lamR));
# printf("%s%8.1f\n","transformations in stomach....",$ST dis);
# printf("%s%8.1f\n","transformations in small Int...",$SI dis);
# printf("%s%8.1f\n","transformations in U.L. Int....",\$UL dis);
# printf("%s%8.1f\n","transformations in L.L. Int....",$LLI dis);
#####-GI activity transported to the transfer compartment-####
```

```
$qi tr=($lamST*$lamBF)/(($lamST+$lamR)*($lamSI+$lamBF+$lamR));
# printf("%s%6.2e\n","GI activity to tr....",$gi_tr );
####-activity in tr (blood) and ti (tissue) compartments-####
 $tr_dis = ($ln_tr + $gi_tr)/($lamTR+$lamR);
# printf("%s%8.0f\n","tr.....",$tr_dis );
 $ti dis = ($F TI * $lamTR * ($ln tr+$qi tr)) /
            (($lamTR+$lamR) * ($lamTI+$lamR));
# printf("%s%8.0f\n","ti.....",$ti dis );
#####end of the activity calculations#####
#
####-calculate doses and print out results here-####
   printf("%24s%6.2e\n","Dose to $trgt_org = ",&ORGANDOSE);
   print OUTPUT sprintf(" %1s %7.3e","=",&ORGANDOSE);
   $effec = $effec + &ORGANDOSE * $org wt{$trgt org} ;
  } #end of clearence loop
   print OUTPUT "\n";
 } #end of main calculational loop
   print OUTPUT sprintf("%9s %1s %6.3e","Effective","=",$effec) ;
   print OUTPUT "\n----\n";
} # end of main input loop
close(OUTPUT);
unless( $! = 0 ) {print "***ingdose.pl ended succesfully***\n"};
sub ORGANDOSE {
 $current_dose
              = $ST_dis
                         * @see[0] +
                         * @see[1] +
                 $SI dis
                 $ULI_dis * @see[2] +
                 $LLI dis * @see[3] +
                 \frac{\overline{d}}{dis} * @see[5] +
                 $ti_dis * @see[4];
 return $current dose; }
```

Air Quality Group

Test case radionuclide:			
Record	source of data	Checked?	Date
Radionuclide.tab file	http://www.nndc.bnl.gov/nndc/nudat/rad		
	<u>form.html</u>		
FGR tables	http://www.ornl.gov/~wlj/fgr12tab.htm		
List of emissions	Screen output from endsf.pl		
Immersion EDE	Screen output from DOSFACTOR code		
Ground surface EDE	Screen output from DOSFACTOR code		
Inhalation EDE	Screen output from inhdose.pl		
Ingestion EDE	Screen output from injdose.pl		
Summary table and notes	Spreadsheet		
Signature Radionuclide:	Name (print) Date		-
Record	source of data	Checked?	Date
Radionuclide.tab file	http://www.nndc.bnl.gov/nndc/nudat/rad		
List of emissions	Screen output from endsf.pl		
Effective dose factor - Immersion	Screen output from DOSFACTOR code		
Effective dose factor - ground	Screen output from DOSFACTOR code		
surface	_		
Summary table and notes-external	spreadsheet		
factors			
Input to ICRP38.DAT	Copy of output file from tab2icrp.pl		
	code		
Summary of Energy Emitted	Screen output from RADSUM code		
Input Data to SEECAL	Output data from DEXRAX		
Summary of SEE calculations	Screen output from SEECAL program		
SEE Values	Copy of output file from SEECAL program		
Metabolic data	Copy of *.bio input file to inhdose.pl		
Inhalation Dose Factor	Copy of output file from inhdose.pl		
Ingestion Dose Factor	Copy of output file from ingdose.pl		
Input file to fgr.pl program	File copy		
Table 2-like Dose Conversion	Screen output from fgr.pl code		-
factor			
Prepared by:			
 Signature	Name (print) Date	<u> </u>	-
Olgrialule	rianie (Pilit) Date	,	

Air Quality Group

QUALITY ASSURANCE REVIEW OF DOSE FACTORS CALCULATED FROM NUCLEAR DECAY DATA

		This form is from ES	H-17-511		
Radionuclide:					
I certify that the data, calculations, fin	al result, and documenta	ation are complete for this radionuclid	le.		
Preparer's signature	Name (print)	Date			
The Preparer's check sheet wa	s completed				
The test case was included					
The correct nuclear decay data	was used				
The Immersion dose factor was cross-checked					
The Ground Surface dose factor was cross-checked					
The Inhalation dose factor was cross-checked					
The Ingestion dose factors was cross-checked					
The Overall dose factor was verified					
I have peer-reviewed the data, calculations.	ations, final result, and d	ocumentation, including the items lis	ted		
Reviewer signature	Name (print)	 			